Software AG Editor Main Commands

This section covers the following topics:

- General Information
- Main Commands Command Syntax
- Common Command Options
- Main Command Descriptions

General Information

The Software AG Editor provides two types of commands for editing:

- **Main commands** (described below).
- **Line commands** (see the section Software AG Editor Line Commands).

This section gives a short description of each standard **main command** and a complete overview of the command syntax.

Main commands are issued from the command line COMMAND===> in the second screen line.

The main commands available to you depend on the application you are using.

Note:
Depending on the configuration of your installation, main commands and line commands may be entered in lower case. In this section, however, all commands are shown in upper case to distinguish them as commands.

Main Commands Command Syntax

CURSOR	Terms in upper-case bold letters are used for commands.
_	Full underlining indicates a default value; partial underlining indicates an abbreviated form.
string	Terms in lower-case italic letters are used for parts of syntax or commands whose values have to be supplied by the user.
[ ]	Optional command operands are enclosed within square brackets. If a choice is offered, one or more operands can be specified.
{ }	Braces indicate that a choice of operands must be made. Only one of the operands enclosed within braces can be supplied.
[{ }]	A combination of square brackets and braces indicates that the enclosed operands are optional, but if you wish to specify one, only one can be selected.
...4	Terms preceding the ellipsis may be repeated. A numeric constant after an ellipsis limits the number of times the term may be repeated.
&&	The Editor provides a command redisplay feature: if you precede a command with two ampersands (&&), it remains displayed in the command line and is executed every time you press **Enter** until you delete the command or overtype it.

Common Command Options

There are some options which are available with several main commands. These options are described below for all commands in which they can be specified.

Line Specifications

With these options, you can restrict the effect of a command to a certain range of lines:

.X	The command affects only the lines from the line labelled ".X" to the last line.
.X .Y	The command affects only the lines from the line labelled ".X"to the line labelled".Y".

X and Y can also be any label of 1 to 4 alphabetical characters (see the LABEL command).

Column Specifications

With these options, you can restrict the effect of a command to a certain range of columns. These column numbers refer to the actual data columns; the line numbers preceding the data are not counted. So, if you specify Column 1 with a command, this may physically be the 8th column of your screen, but it is in fact the 1st column of the data you are editing.

n	The command affects only lines in which the specified *string* begins in Column *n* (that is, the first character of the *string* must be in Column *n*).
n m	The command affects only lines in which the specified *string* occurs anywhere between columns *n* and *m*.

Displayed or Nondisplayed Lines

With one of the following options, you can specify that only excluded or only included lines are to be affected by a command:

NX	The command affects only non-excluded lines; that is, lines which are currently being displayed.
X	The command affects only excluded lines; that is, lines which are currently **not** being displayed as specified by the EXCLUDE command. An excluded line remains excluded from display if a main command function is performed on it.

Direction of Operation

With these options, you can specify the direction in which a command is to operate:

NEXT	The command affects the next line (starting from the cursor position) in which the specified *string* occurs.
PREV	The command affects the line that contains the previous occurrence of the specified *string*.
FIRST	The command affects the first line in which the specified *string* occurs.
LAST	The command affects the last line in which the specified *string* occurs.
ALL	The command affects all lines in which the specified *string* occurs.

Special Occurrences

With these options, you can specify whether only special occurrences of the specified *string* are to be affected by a command:

CHARS	The command affects any line in which the specified *string* occurs.
WORD	The command affects only those lines in which the specified *string* forms a word.
PREFIX	The command affects only those lines in which the specified *string* is the beginning of a word.
SUFFIX	The command affects only those lines in which the specified *string* is the end of a word.

Main Command Descriptions

ADVANCE

This command is used to specify whether the cursor moves to the next line automatically after a line update.

ON	The cursor moves to the next line after an update.
OFF	The cursor does not move to the next line after an update.
PAGE	The line containing the cursor is placed at the top of the edit area after an update.

If an unqualified ADVANCE command is issued, it is interpreted as ADVANCE ON. The default setting is ADVANCE ON and can be changed by editing your profile; see Setting Your Editor Profile.

AORDER

This command is used to specify whether newly-entered text is to be automatically justified within the set boundaries.

If an unqualified AORDER command is issued, it is interpreted as AORDER ON. The base setting can be changed by editing your profile; see Setting Your Editor Profile.

AUTOREN

For PDS members and sequential data sets only. This command is used to specify whether the Editor automatically activates the RENUMBER function.

If an unqualified AUTOREN command is issued, it is interpreted as an AUTOREN ON command. The base setting can be changed by editing your profile; see Setting Your Editor Profile.

AUTOSAVE

This command is used to specify whether the Editor executes an automatic SAVE command when you issue the END command.

If an unqualified AUTOSAVE command is issued, it is interpreted as an AUTOSAVE ON command. Default setting is AUTOSAVE ON and can be changed by editing your profile; see Setting Your Editor Profile.

BNDS

This command is used to restrict the effect of certain commands to a specific range of columns.

These boundaries apply to the main commands FIND, CHANGE, CENTER, ORDER, JLEFT and JRIGHT, and their corresponding line commands (TC, TO, LJ, RJ, etc.).

n	The number of the column at which the left boundary is to be placed.
m	The number of the column at which the right boundary is to be placed.

If *n* and *m* are omitted, the boundaries are set at the first and last column of the edit area.

To see the current boundary settings, issue the BNDS line command.

BOTTOM

This command is used to scroll to the end of the object being edited.

CANCEL

This command cancels all changes made after the last SAVE or STOW command and leaves the Editor.

CAPS

This command is used to switch upper-case translation on and off.

ON	The data is translated to upper case.
OFF	The data is not translated; that is, it remains as entered.
PGM	The data is translated to upper case (except for comments, which remain as entered).

The CAPS command issued without a parameter has the same effect as CAPS ON. The default is CAPS ON. Edit your profile to change this; see Setting Your Editor Profile.

CENTER

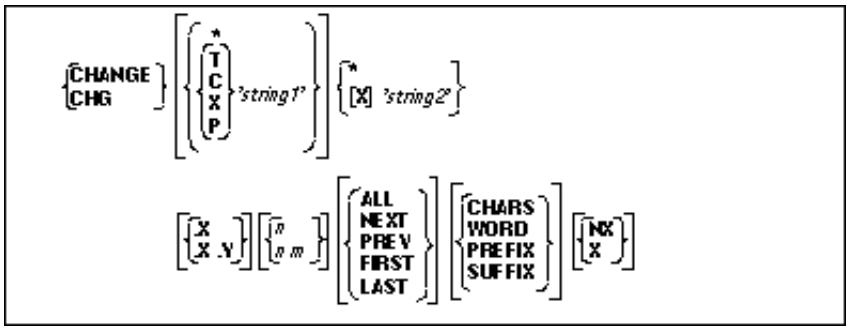
This command is used to center data.

ALL	Centers the data of all lines.
n	Centers the data from line *n* to the last line.
n m	Centers the data from line *n* to line *m*.

The CENTER command applies only within the horizontal boundaries as set with the main command BNDS.

For centering, you can also use the line commands TC and TCC.

CHANGE



This command is used to replace a character string (*string1*) by another character string (*string2*).

You can specify the string to be replaced (*string1*) as follows:

T'*string1*'	Replace *string1* irrespective of whether it occurs in lower case or upper case. This is the default.
'*string1*'	Same as **T'***string1*'.
C'*string1*'	Replace *string1* only if it occurs exactly as specified.
X'*string1*'	Replace the string that corresponds to the specified hexadecimal character string *string1*. Replace it by the hexadecimal string *string2*.
P'*string1*'	Replace *string1* which includes the following wildcard characters: **=** any character § alphabetical character **#** numeric character **$** special character ^ non-blank character **-** non-numeric character **<** lower-case character **>** upper-case character
*****	Use the search string specified in a previous command (for example, FIND, CHANGE, EXCLUDE).

If you want an apostrophe to be part of *string1* or *string2*, you must write it as two apostrophes.

All other options of the CHANGE command are described in the section Common Command Options.

Using CHANGE Together with Other Commands

To repeat the execution of a CHANGE command, you use the command RCHANGE.

To search the entire data for a character string and then decide occurrence by occurrence whether to replace it by another character string, you can use a combination of the commands FIND, CHANGE, RFIND and RCHANGE:

First, you search for the string:

```
FIND 'string'
```

When the string has been found, you can decide whether to:

- replace it: `CHANGE 'string' 'new-string'`
- or search for the next occurrence of the string by repeating the FIND command: `RFIND`

When the next occurrence of the string has been found, you can again decide whether to:

- replace it by repeating the CHANGE command: `RCHANGE`
- or search for the next occurrence of the string by repeating the FIND command: `RFIND`

Examples of the CHANGE Command

Example 1:
```
CHG 'LOW' 'HIGH'
```
This command replaces the first occurrence of "low" by "high" (regardless of upper or lower case).

Example 2:
```
CHG C'OPS' 'SPF' .X .Y 28 32 ALL
```
This command changes "OPS" (exactly as entered here) into "SPF"; it changes all occurrences in the block of lines labelled by ".X" and ".Y" and between columns 28 and 32.

Example 3:
```
CHG C'NAME' 'APPL' .X .Y ALL PREFIX NX
```
This command changes all occurrences of prefix "NAME" (exactly as entered here) into "APPL" in all displayed lines in the block labelled by ".X" and ".Y".

Example 4:
```
CHG * 'NEW'
```
This command replaces the next occurrence of the string specified in the last CHANGE command by the string "NEW".

Example 5:
```
CHG 'OLD' *
```
This command replaces the next occurrence of the string "OLD" by the same new string as specified in the last CHANGE command.

CLEAR

This command clears the edit area of content.

Warning:
This command may result in loss of data. Any changes made since the last save of the object in the edit area are lost. To prevent data loss, issue the SAVE command prior to the CLEAR command.

COLS

This command displays a line at the top of the edit area showing column positions.

To display the column positions, you can also use the line command COLS.

CURSOR

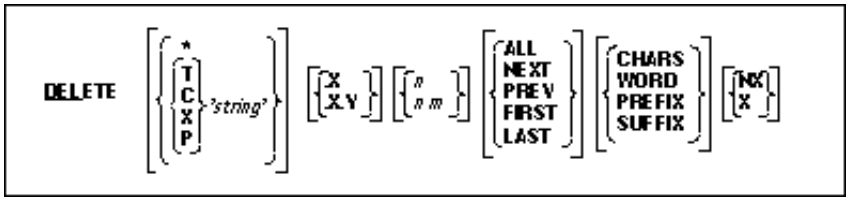
This command returns the cursor to the command field when you next press **ENTER**.

CWINDOW

This command is used to copy a data window according to the command parameters.

l	The number of the line in which the data window is to be inserted.
n	The number of the column in which the data window is to be inserted.

DELETE



This command is used to delete lines.

You can specify that only lines which contain a specified character *string* are to be deleted. You have the following options:

T'*string*'	Delete lines that contain the *string* irrespective of whether it is in lower case or upper case. This is the default.
'*string*'	Same as **T'***string***'**.
C'*string*'	Delete lines that contain the *string* exactly as specified.
X'*string*'	Delete lines that contain the string which corresponds to the specified hexadecimal character *string*.
P'*string*'	Delete lines that contain the *string* which includes the following wildcard characters: = any character § alphabetical character # numeric character **$** special character ^ non-blank character - non-numeric character < lower-case character > upper-case character
*****	Use the search string specified in a previous command (for example, FIND, CHANGE, EXCLUDE).

All other options of the DELETE command are described in the section Common Command Options.

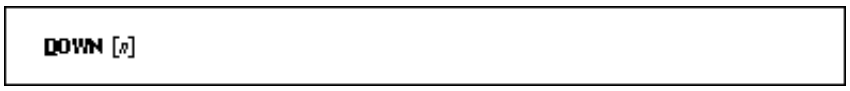
If you enter the DELETE command without any parameters, the current line is deleted.

Example 1:
```
DEL C'NAME' 1 20 ALL PREFIX NX
```
This command deletes all lines that contain the string "NAME" (in upper case exactly as entered here) as a prefix to a word in all lines not excluded from display, if "NAME" occurs between columns 1 and 20.

Example 2:
```
DEL C'Abc' .X .Y 10 30 ALL
```
This command deletes all lines that contain the string "Abc" (exactly as entered here) between columns 10 and 30 within the block of lines labelled by ".X" and ".Y"

To delete lines, you can also use the line commands D, D*n* and DD.

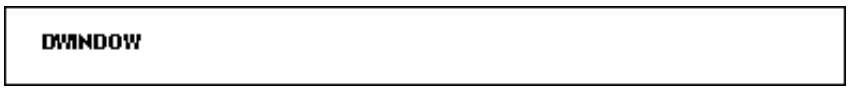
DOWN



This command is used to scroll downwards in the data.

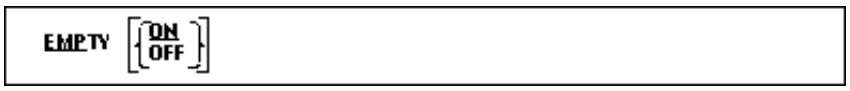
The parameter *n* specifies the number of lines to be scrolled downwards. If *n* is omitted, the scroll amount is determined by the scroll mode.

DWINDOW



This command is used to delete the last defined data window.

EMPTY



This command controls the deletion of blank lines in the Editor.

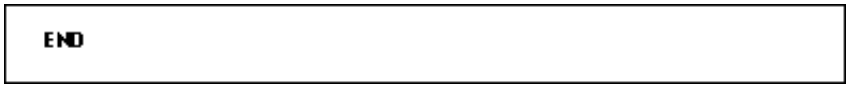
OFF	Empty lines are not deleted.
ON	Empty lines are deleted.

If you enter EMPTY without any parameter, it is interpreted as EMPTY ON. The default setting is EMPTY OFF (no suppression) and can be changed by editing your profile; see Setting Your Editor Profile.

END

Stores the data including all changes and leaves the Editor.

The command format is:



If AUTOSAVE is set to OFF and you have changed data, the Editor asks you to issue the SAVE or CANCEL command.

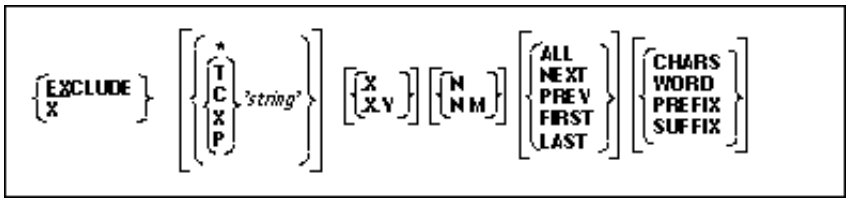
ESCAPE

This command activates/deactivates the escape character to precede line commands entered in the first column of the data.

The parameter **character** is the special character to be used. The default escape character is the period (.).

If you issue the ESCAPE command without any parameter, it is interpreted as ESCAPE ON. Default is ESCAPE OFF.Can be changed by editing profile; see Setting Your Editor Profile.

EXCLUDE



This command is used to exclude lines from being displayed.

You can specify that only lines which contain a specified character *string* are to be excluded from display. You have the following options:

T'*string*'	Exclude lines that contain the *string* irrespective of whether it is in lower case or upper case. This is the default.
'*string*'	Same as **T'***string***'**.
C'*string*'	Exclude lines that contain the *string* exactly as specified.
X'*string*'	Exclude lines that contain the string which corresponds to the specified hexadecimal character *string*.
P'*string*'	Exclude lines that contain the *string* which includes the following wildcard characters: = any character § alphabetical character # numeric character **$** special character ^ non-blank character - non-numeric character < lower-case character > upper-case character
*****	Use the search string specified in a previous command (for example, FIND, CHANGE, EXCLUDE).

All other options of the EXCLUDE command are described in the section Common Command Options.

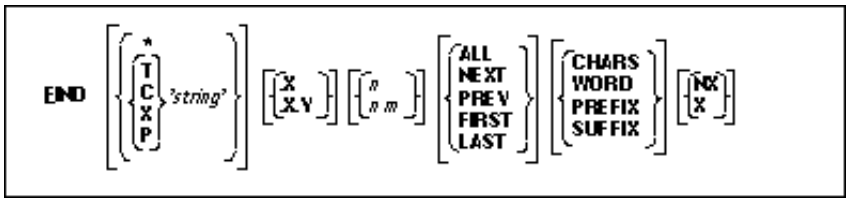
If you enter the EXCLUDE command without any parameters, the current line is excluded from display.

Example 1:
```
EX 10
```
This command excludes line 10 from display.

Example 2:
```
EX C'NAME' .X ALL PREFIX
```
This command excludes from display all lines which contain "NAME" (in upper case as entered here) as a prefix to a word, starting from the line labelled with ".X".

To re-display excluded lines, you use the main command INCLUDE.

FIND



This command is used to search for a specific character *string*. The cursor is placed on the beginning of the first found *string*. If the line containing the *string* was excluded from display, it is displayed when found.

You can specify the *string* as follows:

T'*string*'	Search for the *string* irrespective of whether it is in lower case or upper case. This is the default.
'*string*'	Same as **T'***string***'**.
C'*string*'	Search for the *string* exactly as specified.
X'*string*'	Search for the string that corresponds to the specified hexadecimal character *string*.
P'*string*'	Search for a *string* which includes the following wildcard characters: = any character § alphabetical character # numeric character **$** special character ^ non-blank character - non-numeric character < lower-case character > upper-case character
*****	Search for the *string* specified in the previous FIND command.

If you want an apostrophe to be part of the *string*, you must write it as two apostrophes.

All other options of the FIND command are described in the section Common Command Options.

Example 1:
```
F C'NAME' .X .Y ALL PREFIX X
```
This command searches for any occurrence of "NAME" exactly as entered here as a prefix of a word in any excluded line within the block delineated by ".X" and ".Y".

Example 2:
```
F C'HILITE' X PREV
```
This command searches for the previous occurrence of "HILITE" exactly as entered here in any excluded line.

Example 3:
```
F P'RCV#' .X .Z 20 30
```
This command searches for any 4-character string that begins with "RCV" and whose fourth character is numeric. It searches within the block of lines delineated by ".X" and ".Z" and between columns 20 to 30.

Example 4:
```
F X'6C' SUFFIX NX
```
This command searches for the character that is hexadecimally represented as "6C". Only those occurrences of the character that are at the end of word are found. The search is valid for non-excluded lines only.

Example 5:
```
F ''''w'
```
This command searches for the following character string: **'w**

Example 6:
```
F 'r''w'
```
This command searches for the following character string: **r'w**

Example 7:
```
F ''''
```
This command searches for an apostrophe.

The FIND command differs from the LOCATE command in the following ways:

- The FIND command is more effective for text searches while the LOCATE command is used primarily to find line numbers or line labels.
- The LOCATE command finds only text in upper case beginning in column one of the Editor. In addition, in order to find a string, the data in the Editor must be in alphabetical order.
- When a line is located with the LOCATE command, the cursor is placed in the prefix area and the line is placed at the top of the Editor; with the FIND command, the cursor is placed on the string searched and the line is not necessarily placed at the top of the Editor.

To repeat the execution of a FIND command, use the command RFIND.

FIX

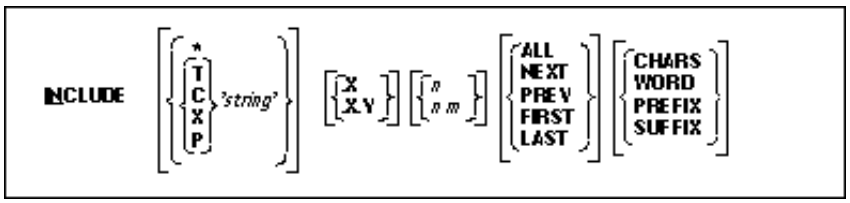
This command is used to specify the number of columns *n*, starting with column 1, to remain in display when scrolling to the right. The default setting is FIX OFF 000 and can be changed by editing your profile; see Setting Your Editor Profile.

HEX

This command is used to switch hexadecimal display mode on and off.

The default setting is HEX ON and can be changed by editing your profile; see Setting Your Editor Profile.

INCLUDE



This command is used to re-display lines that were excluded from display by an EXCLUDE command. The command takes the same parameters as the EXCLUDE command.

If you enter the INCLUDE command without any parameters, it includes the first line of an excluded block.

Example:
```
IN C'NAME' .X ALL PREFIX
```
This command recalls all excluded lines with NAME as a prefix to a word exactly as entered here, starting from the line labelled ***.X***.

JLEFT

This command is used to align data left-justified.

ALL	Aligns the data of all lines.
n	Aligns the data from line *n* to the last line.
n m	Aligns the data from line *n* to line *m*.

The JLEFT command applies only within the horizontal boundaries as set with the main command BNDS.

Example:
```
BNDS 10;JLEFT 15 20
```
The data between column 10 and the rightmost column of your screen in lines 15 to 20 is left-aligned to column 10.

For left-justification, you can also use the line commands LJ and LJJ.

See also the main command JRIGHT.

JRIGHT

This command is used to align data right-justified.

ALL	Aligns the data of all lines.
n	Aligns the data from line *n* to the last line.
n m	Aligns the data from line *n* to line *m*.

The JRIGHT command applies only within the horizontal boundaries as set with the main command BNDS.

Example 1:
```
BNDS 4 40;JRIGHT 6 18
```
The data between columns 4 to 40 in lines 6 to 18 is right-aligned to column 40.

Example 2:
```
BNDS 10;JRIGHT 15
```
The data to the right of column 10 in line 15 is right-aligned to the rightmost column of your editing screen.

For right-justification, you can also use the line commands LJ and LJJ.

See also the main command JLEFT.

JUSTIFY

This command is used to set the justification mode for the line commands TO and TOO.

TO and TOO are used to join data lines with subsequent lines. Both commands apply only within the horizontal boundaries as set with the main command BNDS.

LEFT	The data is aligned to the left boundary.
RIGHT	The data is aligned to the right boundary.
BOTH	The data is aligned to both boundaries.

Example:

With these commands, you set the horizontal boundaries to columns 10 and 60, and activate left-justification:

```
BNDS 10 60;JUSTIFY LEFT
```

When you then mark a line with a TO line command (or a block of lines with two TOO line commands), the data between columns 10 and 60 in the marked line(s) are left-aligned to column 10.

LABEL

This command is used to mark the current line (that is, the line which is currently at top of the edit area) with the specified *.label*.

The *label* may be a string of 1 to 4 alphabetical characters.

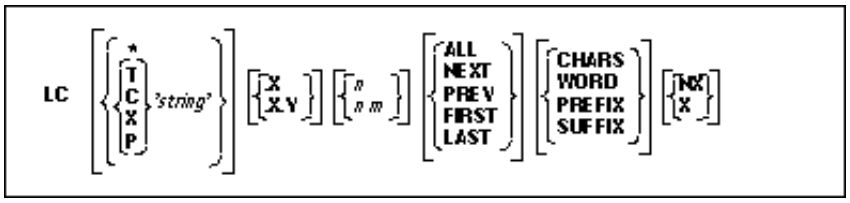
Example:
To label the current line with ".X", you enter the command:

```
LABEL .X
```

You can also mark a block of lines with two labels. For example, to mark a block with labels ".X" and ".Y", you first mark the current line (assuming it is the first line of the block to be marked) with ".X" as shown in the example above; then you scroll until the last line of the block is the current line; then you issue the command "LABEL .Y" to mark that line with ".Y".

To mark a line with a label, you can also use the line command "***.label***"; see the section Software AG Editor Line Commands.

LC



This command is used to change one or more lines to lower case.

You can specify that only lines which contain a specified character *string* are to be changed to lower case. You have the following options:

T'*string*'	Change lines which contain the *string* irrespective of whether it is in lower case or upper case. This is the default.
'*string*'	Same as **T'***string***'**.
C'*string*'	Change lines which contain the *string* exactly as specified.
X'*string*'	Change lines which contain the string that corresponds to the specified hexadecimal character *string*.
P'*string*'	Change lines which contains a *string* that includes the following wildcard characters: = any character § alphabetical character # numeric character **$** special character ^ non-blank character - non-numeric character < lower-case character > upper-case character
*****	Change lines which contain the *string* used in the previous command in which a string was specified.

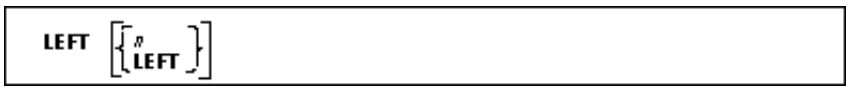
If you want an apostrophe to be part of the *string* , you must write it as two apostrophes.

All other options of the LC command are described in the section Common Command Options.

If you enter the LC command without any parameters, the current line is changed to lower case.

Example:
```
LC C'NAME' .X .Y ALL PREFIX NX
```
This command changes to lower case all displayed lines within the block labelled by ".X" and ".Y" if they contain the string "NAME" (in upper case as entered here) as prefix to a word.

LEFT



This command scrolls the data to the left.

n	Scrolls *n* number of columns to the left.
LEFT	Scrolls the maximum amount to the left.

If *n* or LEFT is omitted, the scrolling amount is determined by the scroll mode.

LIMIT

With this command, you specify the maximum number of lines to be searched with a FIND or RFIND command. The parameter *n* is the number of lines to be searched. This setting can be changed by editing your profile; see Setting Your Editor Profile.

LOCATE

This command is used to scroll a specific line to the top of the edit area (that is, make it the current line).

The command provides the following options:

0	Makes the first line of the data current.
n	Makes line *n* current.
.label	Makes the line labelled **.***label* current.
'string'	Makes the first line that starts with this string current. (The string is only found if it starts in Column 1.)

Examples:

LOC 32	Places line number 32 at the top of the edit area.
32	Same as above
LOC .X	Places the line labelled ".X" at the top of the edit area.

The LOCATE command differs from the FIND command in the following ways:

- The FIND command is more effective for text searches while the LOCATE command is used primarily to find line numbers or line labels.
- The LOCATE command finds only text in upper case beginning in column one of the Editor. In addition, in order to find a string, the data in the Editor must be in alphabetical order.
- When a line is located with the LOCATE command, the cursor is placed in the prefix area and the line is placed at the top of the edit area; with the FIND command, the cursor is placed on the string searched and the line is not necessarily placed at the top of the edit area.

LOG

This command activates or deactivates the log file.

The log file is a history of all modifications made in the Editor since session begin. When the log file is active, each time you press **Enter**, the changes made since the previous **Enter** are recorded in the log file. Using the UNDO command you can consecutively back out changes made since the beginning of the edit session. Edit your profile to change the setting; see Setting Your Editor Profile.

Important: You must ensure the LOG is activated before starting to edit.

MASK

This command activates or deactivates the mask function. When the mask function is active, each time you insert a line in the Editor, a predefined line of text is entered instead of a blank line. The mask line is defined using the MASK line command. The mask function is useful when you must write several lines of code which are identical or very similar.

To define a mask line, type "mask" over any line number in the Editor and press **Enter**. An empty line appears in which you can type your mask. This mask is active until you update the mask with a new mask line or until you deactivate the mask function.

When the mask function is activated using MASK ON, the mask line appears in all lines added with a line insertion operation. Note, however, that any inserted line is deleted at the next press of **Enter** if nothing is added to it.

The command MASK OFF deactivates the mask function but does not delete the contents of the mask line. The default setting is MASK OFF and can be changed by editing your profile; see Setting Your Editor Profile.

For further information, see Define a Mask Line.

MSO

This command is used to specify whether multiple session operations are possible or not. A multiple session operation is an operation in which data are exchanged between two editing sessions, for example in copying text from one program to another in split-screen mode.

MWINDOW

This command is used to move a data window according to the command parameters.

l	The number of the line in which the data window is to be inserted.
n	The number of the column in which the data window is to be inserted.

NULLS

This command is used determine if the data lines are to be filled with null characters.

ON	The end of each line is filled with null characters.
OFF	Lines are not filled with null characters.

The default setting is NULLS ON and can be changed by editing your profile; see Setting Your Editor Profile.

ORDER

This command is used to join data lines.

ALL	Joins all lines.
n	Joins the lines from line *n* to the last line.
n m	Joins lines from line *n* to line *m*.

The ORDER command applies only within the horizontal boundaries as set with the main command BNDS.

Within the set boundaries, the lines are concatenated and are filled to the greatest possible extent; words that do not fit into one line are automatically placed in the next line.

To join data lines, you can also use the line commands TF, TO and TOO.

POWER

This command switches the Editor to text-entry mode. You are presented with a blank screen into which you can enter one or more lines of text. After entry, press **Enter** and the text is inserted into the first line of the edit area.

PROFILE

This command displays your Editor profile at the top of the edit screen.

With *n* you specify additional lines to be displayed. Possible values for *n* are:

6	Displays your Editor profile and all tab positions (as specified by TABS command).
7	Displays same as **6**, plus the mask line (as specified by the MASK command).
8	Displays same as **7**, plus boundaries (as specified by the BNDS command).
9	Displays same as **8**, plus column numbers (as specified by the COLS command).

PROTECT

This command is used to protect the prefix area (line numbers). To enter line commands with the prefix area protected, type the line command in Column 1 of the edit area preceded by the escape character.

INS	Protects the prefix area of lines added using the insert line command.
ON	Activates protection.
OFF	Deactivates protection.

The default setting is PROTECT ON and can be changed by editing your profile; see Setting Your Editor Profile.

RCHANGE

This command repeats the last CHANGE command.

RECOVERY

This command is used to activate or deactivate the recovery feature for the current edit session. You can also specify the number of updates to be performed before a checkpoint save is performed.

Using parameter n, you specify the number of updated lines after which a checkpoint save is performed.

The default setting is RECOVERY ON and can be changed by editing your profile; see Setting Your Editor Profile.

RENUMBER

For PDS members and sequential data sets only. Specifies renumbering of the lines in the edit area according to the parameters.

ON	Activates renumbering.
OFF	Deactivates renumbering.
n1	Increment of numbering (default is in your edit profile).
n2	Starting column for the new line number (default: 73).
n3	End column for the new line number (default: 80).

To deactivate line renumbering, see the UNREN command.

RESET

This command resets all pending line commands and deletes all line labels.

RFIND

This command repeats the last FIND command.

RIGHT

This command scrolls data to the right.

n	Scrolls *n* number of columns to the right.
RIGHT	Scrolls the maximum amount to the right.

If *n* or RIGHT is omitted, the scrolling amount is determined by the scroll mode.

SAVE

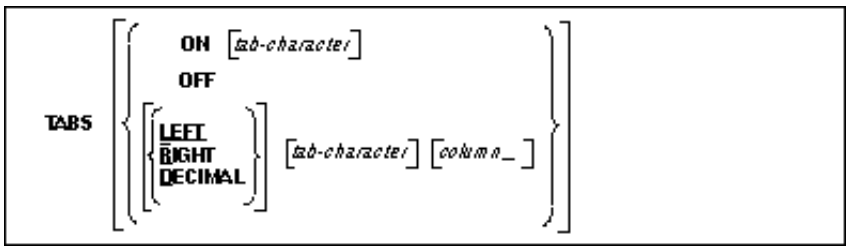
Saves the content of the current Editor session. Syntax and other functionalities depend on the application you are using.

SORT

The SORT command sorts lines in the Editor in ascending or descending alphabetical order. An unqualified SORT command sorts all data in the object in ascending order.

n m	Sorts from Column *n* to Column *m*.
.X	Sorts from line labelled .X to end of object.
.X .Y	Sorts from line labelled .X to line labelled .Y (where .X and .Y are any string of up to four characters).
A	Sorts data in ascending order (A to Z).
D	Sorts data in descending order (Z to A).

TABS



This command is used to control tabulator settings.

You can enable or disable logical or physical tabulation using the command TABS ON/OFF. Tabulation is also enabled by any command that changes a tabulation setting.

For example, the following command enables logical tabulation with the ampersand sign (&) as logical tabulation character:

```
TABS &
```

You set tab positions using the TABS command. For example, the following command sets tabs in Columns 10, 20 and 30:

```
TABS 10 20 30
```

You can enter data and automatically move them to a specific tab position by preceding them with a logical tabulation character. One tabulation character moves the data to the next tab position, two tabulation characters move the data to the second tab position, etc.

To display the current TABS command settings, issue the main command PROFILE.

To display the current tab positions, issue the line command TABS.

The default setting is TABS OFF blank and can be changed by editing your profile; see Setting Your Editor Profile.

Apart from tab positions, you can specify the following parameters with the TABS command:

LEFT	Places the data left-justified at the tab position.
RIGHT	Places the data right-justified at the tab position.
DECIMAL	Places the data so that the decimal point in the data is at the tab position.

To tabulate data in a specific column, multiple tab characters are possible: issue the TABS line command and type over each asterisk (*) marking the tab positions with another special character. Any input preceded by any of these special characters are tabulated in the corresponding column. You can type an L(EFT), R(IGHT) or D(ECIMAL) after each tabulation character to specify placement of data for the tab position.

Examples of the TABS Command

The following examples assume the ampersand (&) to be the tabulation character:

Example 1 - Tab Positions:

The command

```
TABS 10 20 40 LEFT
```

activates logical tabs with tabulation columns 10, 20, and 40 with left justification. After you press **Enter**, the input text line

```
&abc &def &ghi
```

is displayed as follows:

```
  =cols> ----+----1----+----2----+----3----+----4----+----5----+----6
                 abc       def                ghi
```

Example 2 - TABS RIGHT:

The command

```
TABS RIGHT
```

activates logical tabs with right justification. After you press **Enter**, the input text line

```
&abc &def &ghi
```


is displayed as follows:

```
  =cols> ----+----1----+----2----+----3----+----4----+----5----+----6
               abc       def                ghi
```

Example 3 - TABS DECIMAL:

The command

```
TABS DECIMAL
```

activates logical tabs with justification of the decimal point in the tab position. After you press **Enter**, the input text line

```
&15.27$ &16.3 DM &13 IS
```

is displayed as follows:

```
  =cols> ----+----1----+----2----+----3----+----4----+----5----+----6
               15.27$    16.3 DM             13 IS
```

Example 4 - Mixed Justification:

Issue the command **TABS 10 20 30 40 50**. Then issue the TABS line command. This displays the current tab positions as follows:

```
  =tabs            *         *         *         *         *
```

Type an L, R or D next to each tab position as required (unmarked tab positions assume the value of the last TAB command):

```
  =tabs            *R        *D        *D        *D        *L
```

After you press **Enter**, the input text line

```
  &start &0.01 &0.02 &0.03 &end
```

is displayed as follows:

```
  =cols> ----+----1----+----2----+----3----+----4----+----5----+----6
              start        0.01      0.02      0.03       end
```

Example 5 - Multiple Tab Symbols:

Type over the asterisks in the **=tabs** line with other special characters and specify left justification for each one as follows:

```
  =tabs            ]L        &L        #L        $L        =L
```

After you press **Enter**, the input text line

```
  =first$second#third&fourth]fifth
```

is displayed as follows:

```
=cols> ----+----1----+----2----+----3----+----4----+----5----+----6
                                                          first
                                                 second
                                        third
                               fourth
                      fifth
```

Example 6 - Using a Blank as Tabulation Symbol:

Issue the command

```
TABS ' '
```

which activates tabulation with one blank as tabulation character. This means that words separated by one blank are tabulated. After you press **Enter**, the input text line

```
this is a blank tabulation
```

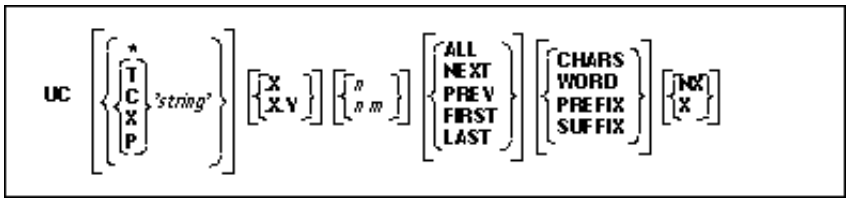
is displayed as follows:

```
=cols> ----+----1----+----2----+----3----+----4----+----5----+----6
               this      is        a         blank     tabulation
```

TOP

This command is used to scroll to the beginning of the object being edited.

UC



The UC command converts one or more lines to upper case. It applies the same parameters as the LC command. If you enter the UC command without parameters, it changes the current line to upper case.

UNDO

If the log file is active (see the LOG command), the UNDO command backs out all changes made since the last time you pressed **Enter**. Repeated use of the UNDO command backs out consecutive changes in reverse order. You can thus back out all changes one by one until you restore the member to its original status at session begin.

You can specify the following parameters with the UNDO command:

ALL	All modifications made in the current edit session are backed out.
n	The last *n* modifications are backed out.

UNREN

Deactivates the renumbering of lines.

n	Specifies the starting column of the line numbers (default: 73).
m	Specifies the end column of the line numbers (default: 80).

To activate line renumbering, see the RENUMBER command.

UP

This command scrolls upwards in the data.

The parameter *n* specifies the number of lines to be scrolled upwards. If *n* is omitted, the scroll amount is determined by the scroll mode.

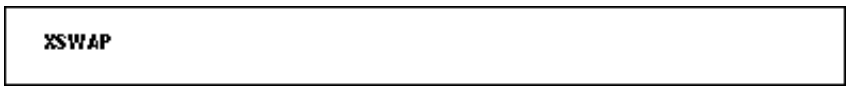
WINDOW

This command is used to define a data window to be copied or moved. The starting line and column and the end line and column of the window are specified in the command parameters. At least *l1* and *l2* are required.

l1 l2	Defines a window starting at Column 1 of Line *l1* and ending in the last column of Line *l2*.
l1 l2 n	Defines a window starting at Column *n* of Line *l1* and ending at the last column of Line *l2*.
l1 l2 n m	Defines a window starting at Column *n* of Line *l1* and ending at Column *m* of Line *l2*.

Note that all data in the data area within the specified points become part of the window. For an example, see the section Copy a Window with Data.

XSWAP



The command is used to exchange displayed lines with excluded lines. Lines are excluded using the EXCLUDE command.

